第4章

FPGA開発ツールの使いかた

――XC3S50活用チュートリアル

西村憲二

ここでは，付属FPGA基板の活用に不可欠なFPGA開発ツール「ISE WebPACK 6.3i」の使いかたを解説する．本誌付属のCD-ROMからツールをインストールし，XC3S50向けの設計を行う．実際に行う手順を具体的に示すので，同じように操作すればFPGA設計を体験できる．（編集部）

付属FPGA基板の設計にはFPGA/CPLD開発ツールが必要になります．付属CD-ROMに収録されている米国Xilinx社の「ISE WebPACK」は，回路図/HDL入力から，論理合成，タイミング解析，論理/タイミング・シミュレーション，FPGA/CPLDへの回路データの書き込みまで，FPGA/CPLD設計に必要なひと通りの機能を備えています．ここでは本ソフトウェアのインストール方法と基本操作について説明します．

表1　ISE WebPACKの動作環境

OS	Windows 2000（SP2以降） Windows XP（SP1以降）
メイン・メモリ	128Mバイト以上
ハード・ディスク	207Mバイト以上の空き容量（インストールのみ）
その他	●CD-ROMドライブ ●パラレル・ポート（Parallel Cable IVを使用する場合） ●インターネット接続環境 ●Netscape 4.7以降，Internet Explorer 5以降のWebブラウザ ●電子メール環境

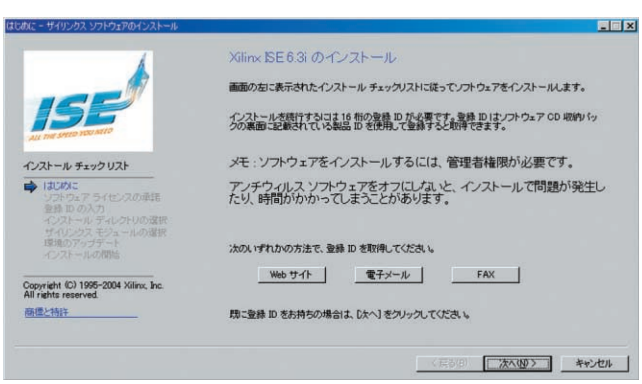


図1　ISE WebPACKのインストール画面

1. ISE WebPACKのインストール

ISE WebPACKは，**表1**に示すパソコン環境で動作します．

付属CD-ROMは2枚あります．ラベルが青色のCD-ROM（Disc 2）にISE WebPACKとISE WebPACKのサービスパックが収録されています．また，ラベルが赤色のCD-ROM（Disc 1）には本チュートリアルで利用できる設計データやSpartan-3のドキュメント類が収録されています．

● ISE WebPACKをインストールする

ISE WebPACKのインストールは，インターネットを使える状態で行うことをお勧めします．また，インストールには，Windows 2000/XPのAdministrator権限が必要になります．会社で支給されているパソコンなどで，Administrator権限がない場合は，パソコンの管理者の方に問い合わせてください．

ラベルが青色のCD-ROMをパソコンのCD-ROMドライブに挿入し，\soft\iseにあるsetup.exeを実行します．すると，**図1**のようにインストーラが起動します．

1）ライセンスの入手

ISE WebPACKは無償で利用可能ですが，ライセンスを取得する必要があります．Product ID（CD-ROMやFPGA基板が納められていた紙製のパッケージにはられているシールに記載）を使用し，インストール用のソフトウェア登

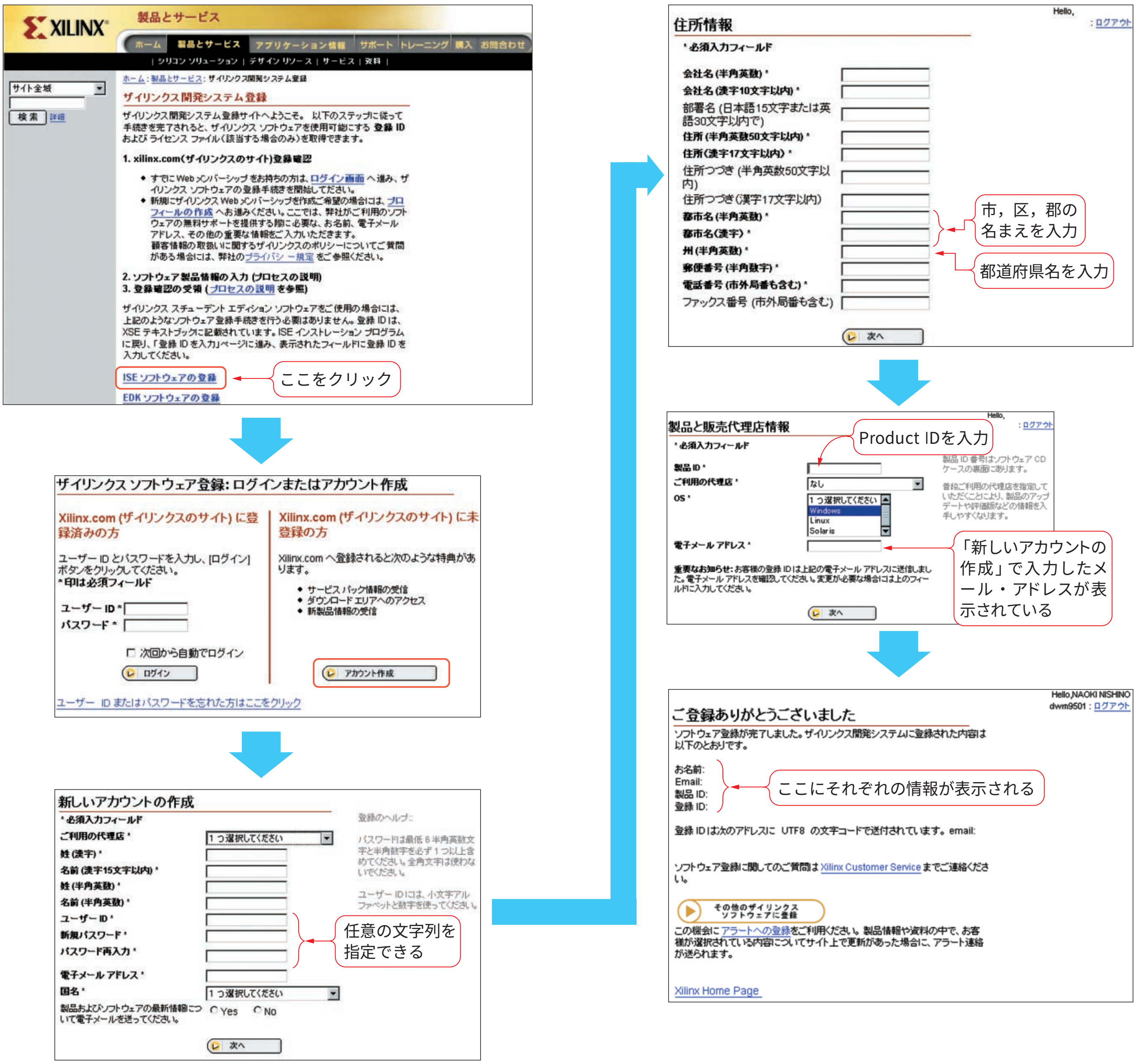


図2　ライセンスの入手

録ID（インストールID）をWebサイトで入手する必要があります注1．

図1の画面で，［Webサイト］ボタンをクリックすると，ユーザ登録のためのWeb画面が開くので，図2の手順でユーザ登録を行います．

まず，「ISEソフトウェアの登録」をクリックします．「ログインまたはアカウント作成」の画面が開きます．すでにユーザIDを持っている方は，IDとパスワードを入力して［ログイン］ボタンを押して作業を進めてください（この場合は図2の「住所情報」のページに進む）．未登録の方は，［アカウント作成］ボタンを押して，以下の作業を行ってください．「新しいアカウントの作成」と「住所情報」の画面で，必要事項を入力してください．ユーザIDは，半角英小文字と数字で任意の文字列を指定できます．ただし，希望するIDがすでに使われている場合は登録できません．エラーになったら変更してください．「製品と販売代理店情報」のページにある「製品ID」には，今回はCD-ROMが納めら

注1：電子メールやFAXなどを使うこともできるが，もっとも簡単で迅速なWebサイトの利用を推奨する．電子メールやFAXによる方法については，ここでは省略する．

4



図3 ソフトウェアのインストール

れていたパッケージにはられているProduct IDを入力してください．Product IDは，画面の説明と異なるので注意してください．ほかの項目についても入力して[次へ]をクリックすると，登録IDが表示されるので，かならず控えておいてください．この情報は，電子メールでも送信されます（電子メールは日本語で送信されるが，文字コードがUTF8のため，電子メール・ソフトウェアによっては文字化けする場合がある）．

2）ソフトウェアのインストール

図1の画面に戻り，[次へ]をクリックし，図3の手順でソフトウェアのインストールを進めます．

まず，ソフトウェアのライセンスの承諾ページが表示されるので，その内容を確認し，承諾するなら「ソフトウェア ライセンスの条項を承諾する」をチェックして[次へ]をクリックしてください．ライセンスを承諾しないと，開発ツールを使うことはできません．次は登録IDの入力画面になります．先ほどのユーザ登録によって入手した登録IDを入力し，[次へ]ボタンを押してください．次の画面でインストールするディレクトリ名とWindowsのメニュー・バーに表示される名まえを入力します．ここはデフォルトのまま[次へ]ボタンを押して次に進むとよいでしょう．インストールするモジュールとデバイス・ファミリの選択では，すべての項目を選ぶことをお勧めします．ISE WebPACKは，付属FPGA基板に搭載されているSpartan-3だけでなく，高性能FPGAのVirtexファミリやCPLDの設計にも使用可能です（高性能ファミリの一部を除く）．環境のアップデートは，デフォルト（2ヵ所ともチェックの状態）を推奨します．インストールの開始画面で，設定した内容を確認し，[インストール]ボタンを押すと，ソフトウェアのインストールが始まります．

インストールの最後に，いくつかの注意点や確認事項が表示されます．これが表示された後，再起動を行ってください．

3）サービスパックのインストール

ソフトウェアのサービスパックが，Xilinx社から定期的に提供されています．ラベルが青色のCD-ROMには，サービスパック1（Xilinx Service Pack 6.3.02i）が収録されています．ISE WebPACKをインストールした後に，サービスパックをかならずインストールしてください．

青色のCD-ROMの\soft\ise-spにある6_3_02i_pc.exeを実行し，図4のような手順でインストールを行います．ディレクトリの選択では，先ほどISE WebPACKをインストールしたディレクトリが表示されていることを確認してください．また，Create Backup before installing updateがチェックしてあることを確かめてから[次へ]ボタンを押します．バックアップ完了の確認後，インストール開始画面で[Install]ボタンを押してください．ファイルのコピーが実行され，いくつかの確認事項が表示された後に，再起動を行ってください．

2. ISE WebPACKを使ってみよう

ここでは，ISE WebPACKを使ったFPGA設計について解説します．論理合成，配置配線，FPGAへのダウンロードについての基本的な機能の使いかた，およびArchitecture WizardとPACEの機能を説明します．Architecture Wizard

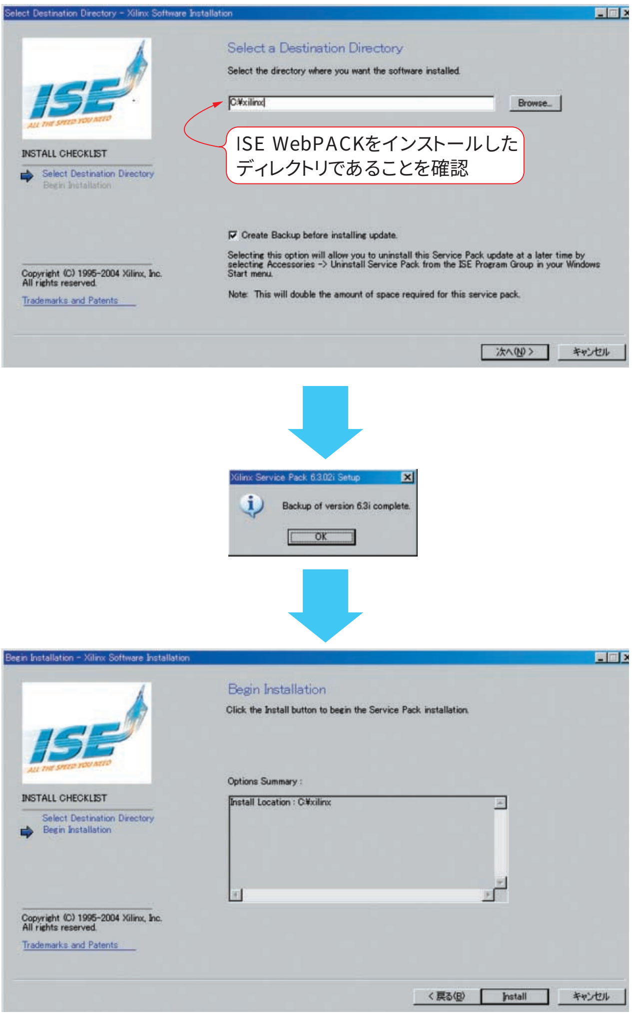
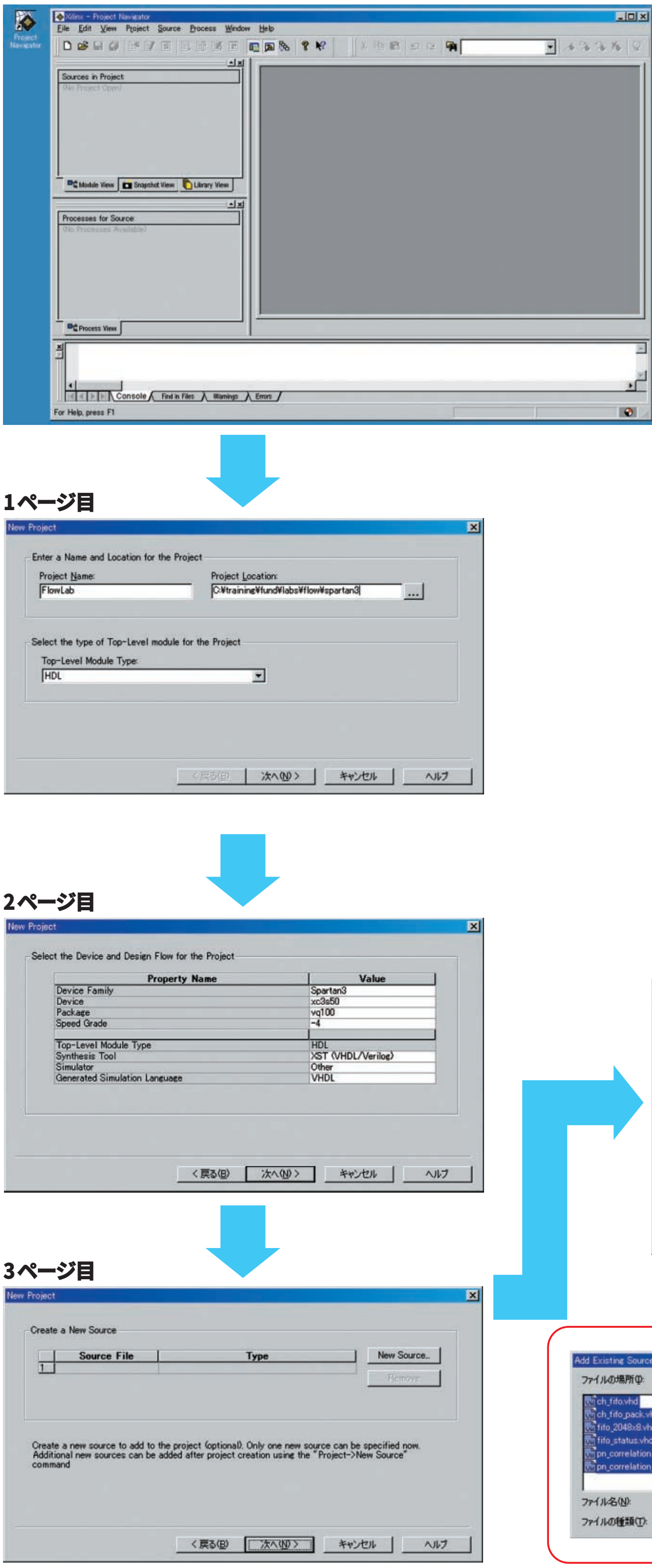


図4　サービスパックのインストール

は，DCM（Digital Clock Management）コンポーネントの生成に使います．また，PACEを使うと，ピンの割り当ておよびフロアプランを行うことができます．

説明に使う設計データは，ラベルが赤色のCD-ROMに収録してあります．このCD-ROMには，設計データとしてVHDLとVerilog HDLの両方のソース・ファイルが収録されていますが，本稿ではVHDLを例にとって説明していきます．「本誌記事関連データ」の中から本稿（2005年1月号特集1の第4章）のページを開き，「現在のフォルダ」にあるtrainingフォルダの中から使用する設計データ（vhdlフォルダのソース・ファイル）をパソコンのハード・ディスクにコピーして使用してください．ここでは，以下のディレク



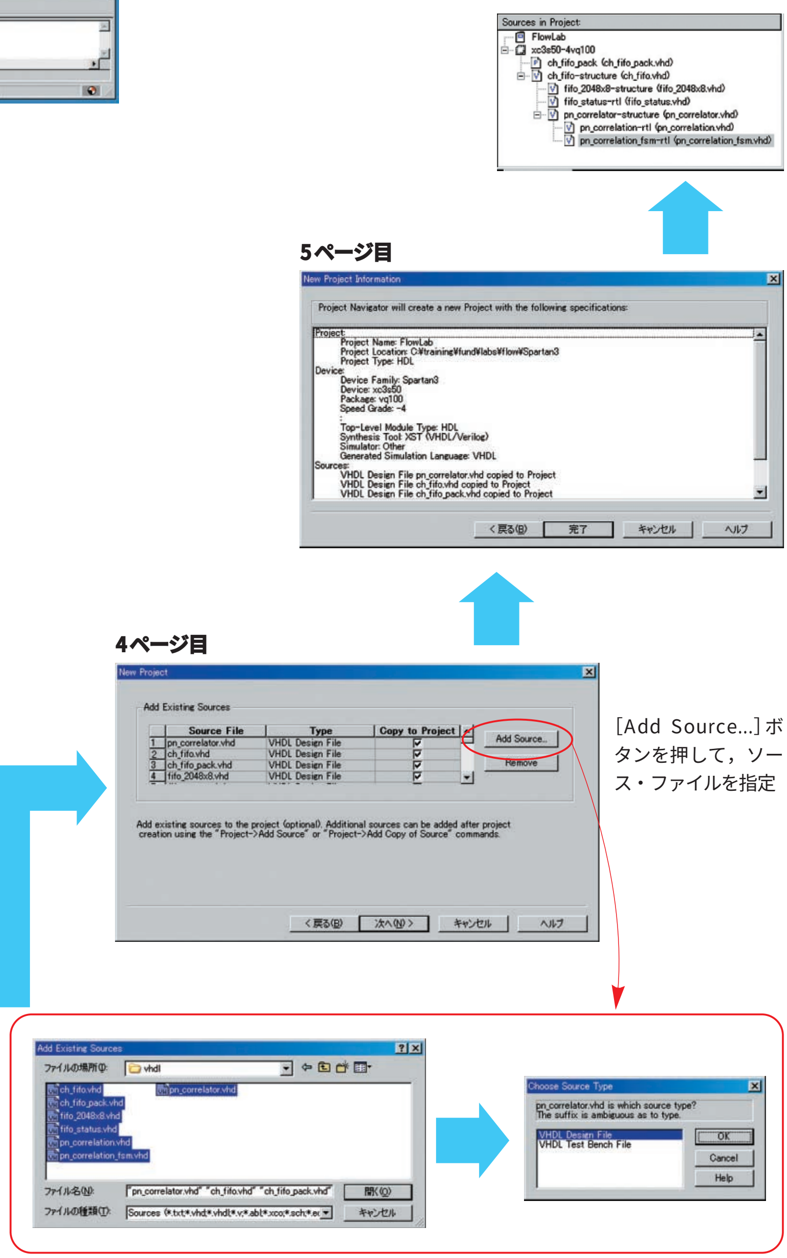


図5　新規プロジェクトの作成

トリにvhdlフォルダごとコピーしたものとして説明します．

C：\training\fund\labs\flow\spartan3

サンプル設計は，階層化されたVHDLファイルとして作成されています．トップのファイルはch_fifoです．fifo_status，fifo_2048x8，my_dcm，pn_correlatorの四つのコンポーネントから構成されています．

● 新規プロジェクトを作成する

デスクトップ上の「Project Navigator」のアイコンをダブルクリックするか，Windowsのスタート・メニューから「プログラム」→「Xilinx ISE 6」→「Project Navigator」でISE WebPACKを起動します．図5の手順で，プロジェクトという設計データを管理するファイルを作成します．

Project Navigatorで「File」→「New Project」を選択すると，New Projectダイアログ・ボックスが開きます．

1ページ目ではプロジェクト名や作業用ディレクトリの指定を行います．Project Name欄には，プロジェクト名として「FlowLab」と入力します．Project Location欄には，作業用ディレクトリ名を指定します．「C：\training\fund\labs\flow\spartan3」と入力するか，ボックスの横にある［...］ボタンをクリックしてディレクトリを選択します．

2ページ目では，ターゲット・デバイスと設計ツールを指定します．Device and Design Flow欄のDevice Familyは「Spartan3」，Deviceは「xc3s50」，Packageは「vq100」，Speed Gradeは「-4」と設定します．また，Synthesis Toolを「XST（VHDL/Verilog）」とします．Generated Simulation Languageでは，シミュレーションで使用する言語を選択します．今回は，「VHDL」とします．このオプションは，シミュレーションだけでなくArchitecture Wizardで生成される言語の選択にも使用するので，設計で使用する言語を選択してください．

3ページ目のCreate a New Sourceは，新たに最上位HDL モジュールを作成する場合の指定です．ここではすべてのソース・ファイルがあらかじめ作成されているので，［次へ］をクリックします．

4ページ目では，使用するソース・ファイルを指定します．作業用ディレクトリにあるすべてのHDLファイルをFlowLabプロジェクトに追加します．まず最初に，［Add Source...］ボタンをクリックし，C：\training\fund\lab\

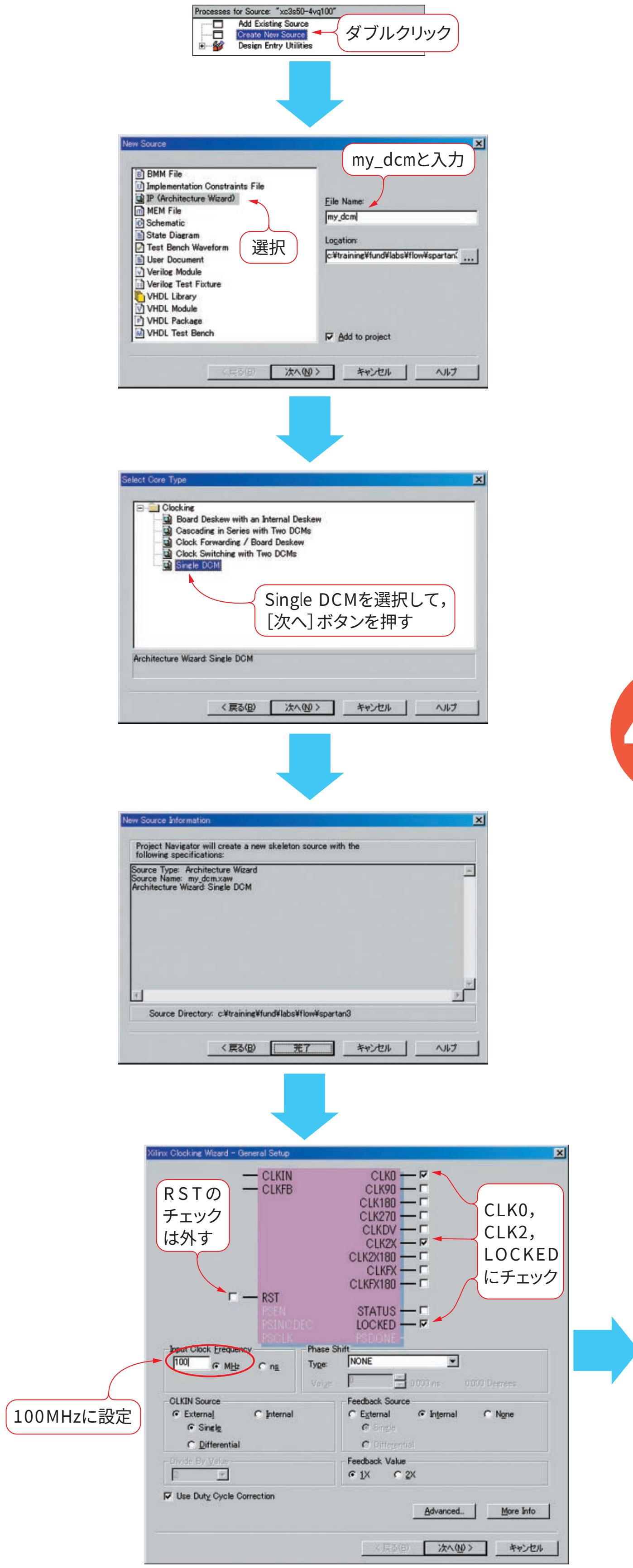


図6　DCMテンプレートの設計（つづく）

flow\spartan3\vhdlにあるすべてのファイルを選択します．

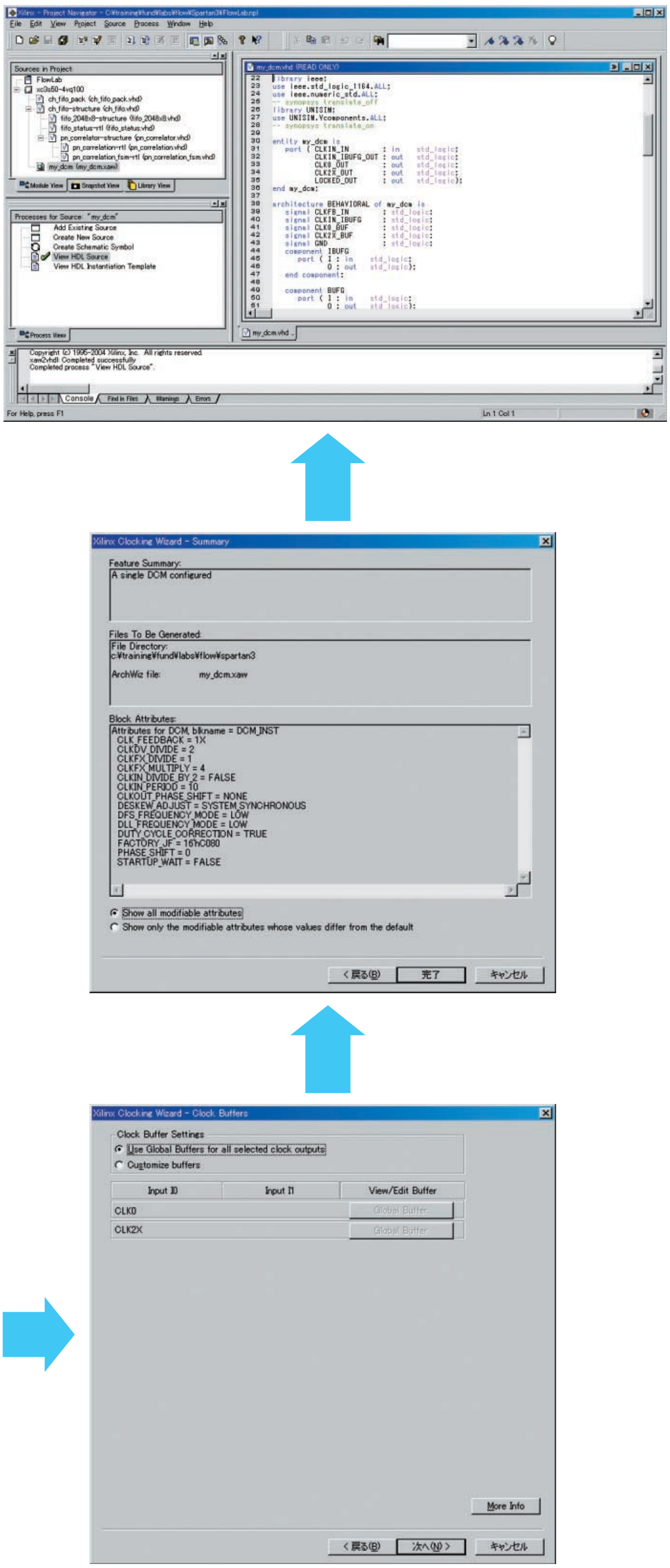


図6　DCMテンプレートの設計（つづき）

Choose Source Typeウィンドウが開くので，追加する各ファイルに対して，設計ファイルかシミュレーション用のテストベンチかを選択します．ここでは，「VHDL Design File」を選択します．

［次へ］ボタンを押すと，5ページ目に設定した内容が表示されるとともに，すべてのファイルが解析されます．［完了］ボタンで階層が自動的に決定され，Sources in Projectウィンドウにファイルが追加されます．

● Architecture WizardでDCMを設計する

Architecture WizardのClocking Wizardを使用して，CLK0，CLK2X，LOCKEDの各ピンを持つmy_dcmという新規ソース・ファイルを作成します．クロック入力周波数は100MHzに設定し，そのほかのオプションはすべてデフォルト設定を使用します．

1）テンプレートの作成

初めにDCMのテンプレートを作成します（図6）．

Processes for Sourceウィンドウで「Create New Source」をダブルクリックします．

New Sourceウィンドウが開くので，「IP（Architecture Wizard）」を選択し，File name欄に「my_dcm」と入力します．次のSelect Core Typeウィンドウでは，［＋］をクリックしてClockingを展開し，「Single DCM」を選択して次に進みます．設定した情報を確認し，［完了］ボタンを押します．

すると，Xilinx Clocking Wizardが開きます．General Setupのページでは，次のオプションを設定します．

- CLK0，CLK2X，LOCKEDの各ピンをチェック
- RSTピンのチェックを外す
- Input Clock Frequency（入力クロック周波数）を100 MHzに設定

ほかの項目はデフォルトのままにします．

DCMには高周波数モードと低周波数モードがあり，どちらを使用するかを決定するために周波数を指定します．入力周波数を入力すると，DCM Wizardで出力周波数が自動的に計算され，その値が指定範囲内であるかどうかが確認されます．また，CLKFX出力を使用する場合，入力周波数を使用してCLKFXの出力周波数およびジッタが計算されます．

すべての設定が終了したら，［次へ］をクリックします．

Clock Buffersページのオプションはすべてデフォルトの

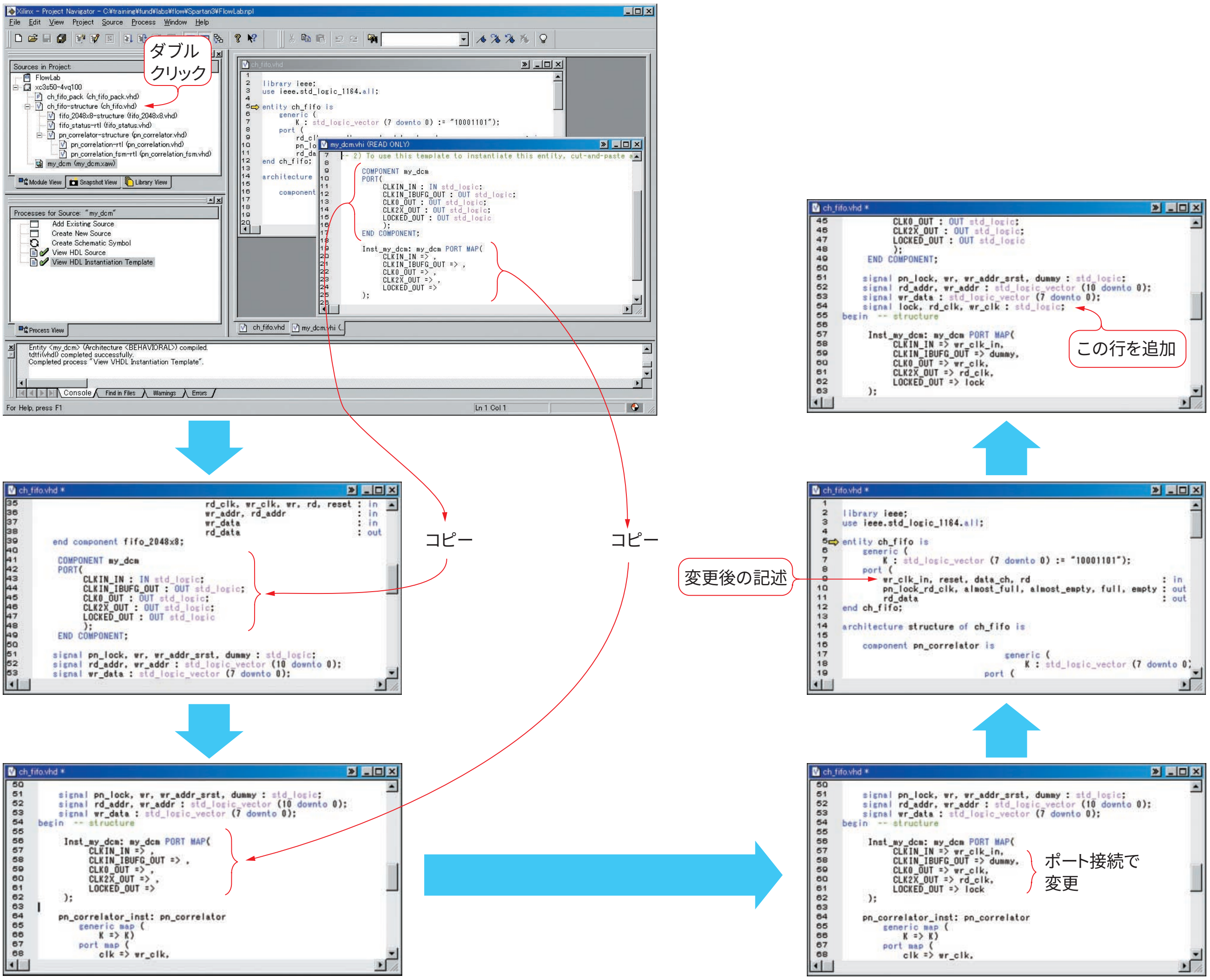


図7　DCMのデザインへの追加

ままにしておいて，次に進みます．設計内容が表示されたら，［完了］をクリックします．Sources in Projectウィンドウに新しいファイル（my_dcm.xaw）が追加されます．my_dcm.xawファイルを選択し，Processes for SourceウィンドウでView HDL Sourceをダブルクリックして Architecture Wizardで生成されたソース・コードを確認します．

2）デザインへの追加

次に，生成したDCMをch_fifo.vhdに組み込みます．テンプレートからコンポーネント宣言およびコンポーネント・インスタンスをコピーし，組み込もうとするソース・コードにはり付けます．その後，信号を割り当て，コードを変更します（図7）．

まず，Sources in Projectウィンドウで「ch_fifo.vhd」をダブルクリックしてソース・コードを開きます．

次に，Sources in Projectウィンドウで「my_dcm.xaw」を選択し，Processes for Sourceウィンドウで「View HDL Instantiation Template」をダブルクリックして，テンプレート・ファイルを開きます．テンプレート・ファイルが開かない場合は，再度ダブルクリックしてください．

テンプレートから，コンポーネント宣言（`COMPONENT my_dcm … END COMPONENT;`）をコピーし，ch_fifo.vhdの40行目（`end component fifo_2048x8;`の後）にはり付けます．また，コンポーネント・インスタンス（`Inst_my_dcm;my_dcm PORT MAP(…);`）をコピーし，ch_fifo.vhdの`begin -- structure`の後にはり付けます．

4

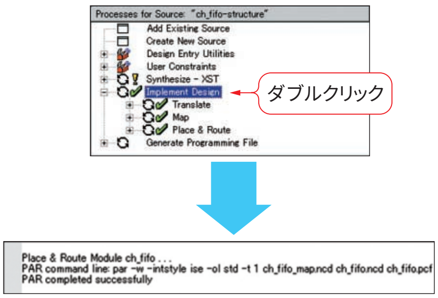


図8　インプリメンテーション（論理合成，配置配線）

ポートの接続で，次のように変更します．

```
Inst_my_dcm:my_dcm PORT MAP(
      CLKIN_IN => wr_clk_in,
      CLKIN_IBUFG_OUT => dummy,
      CLK0_OUT => wr_clk,
      CLK2X_OUT => rd_clk,
      LOCKED_OUT => lock,
);
```

ch_fifoには，二つの入力クロック（wr_clkとrd_clk）があります．ここでは，作成したDCMから二つのクロックを合成するようにコードを変更します．DCMインスタンスを機能させるには，次の手順に従ってコードを編集してください．

- ch_fifoエンティティ宣言の中のrd_clk信号を削除する（この信号はDCMで生成される）．wr_clkをwr_clk_inに変更する．
- `signal lock,rd_clk,wr_clk : std_logic;`を追加する．

最後に，「File」→「Save」でファイルを保存します．

● 論理合成，配置配線を行う

設計ファイルの論理合成と配置配線を行い，FPGAに実装可能なデータに変換します（図8）．

Sources in Projectウィンドウで最上位ファイルのch_fifo.vhdを選択し，Processes for Sourceウィンドウで「Implement Design」をダブルクリックします．これで，必要なすべての処理が実行されます．

Projectes Navigatorの右下に進行状況を示すバーが表示されます．Processes for Sourceに表示されるマークとログ・ウィンドウに表示されるメッセージを確認します．Processes for SourceウィンドウでImplement Designの横にある［+］をクリックすると，詳細な進行状況が表示されます．処理が終了すると，次のいずれかのマークが横に示されます．

- 緑のチェック・マーク：処理の正常終了
- 黄色の感嘆符：警告あり
- 赤の×マーク：エラー発生

警告が表示されることがありますが，今回は無視しても問題ありません．

処理が終了すると，ログ・ウィンドウに次のようなメッセージが表示されます．

```
PAR completed successfully
```

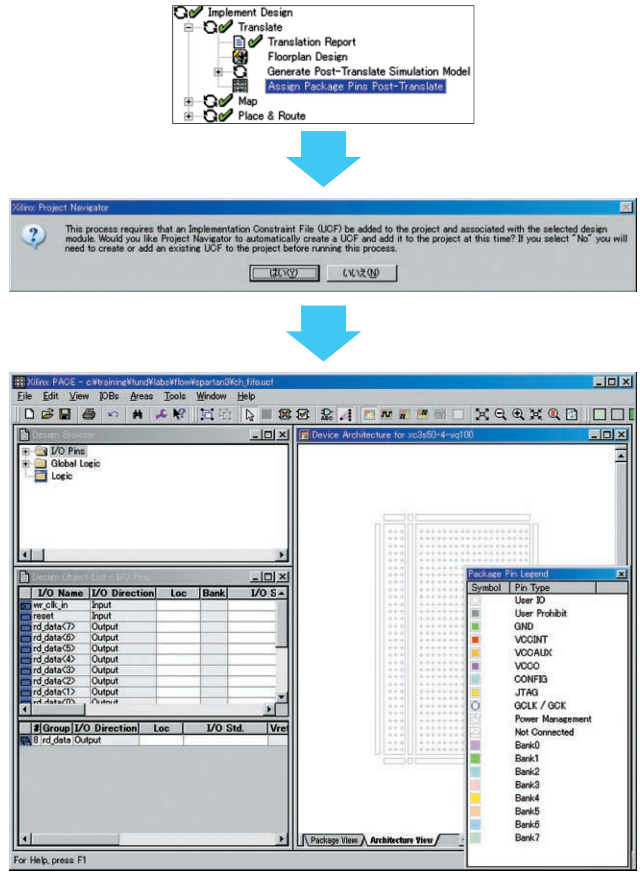
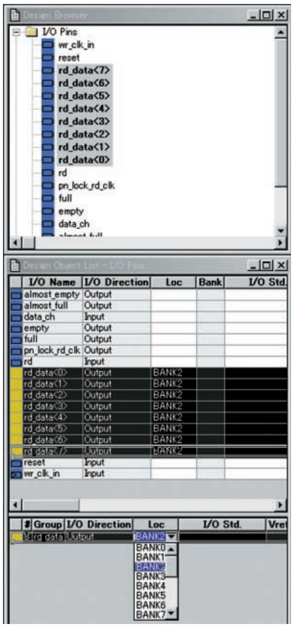


図9　PACEの起動

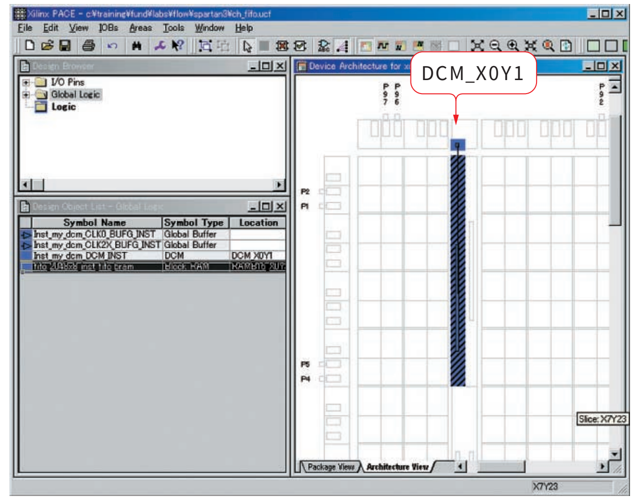
● ピン配置とフロアプランを表示する

ピン配置やフロアプランの表示を行うためには，PACEというツールを利用します（図9）．

Processes for Sourceウィンドウで「Implementation Design」の下の「Translate」を展開し，「Assign Package



◀図10
ピン配置
(GUIによるバンク指定)



▶図11
フロアプラン

Post-Translate」をダブルクリックします．新しい制約ファイルを作成するかどうか聞かれるので，［はい］をクリックします．これでch_fifo.ucfという制約ファイルが作成され，PACEが起動します．

1）ピン配置（GUIによるバンク指定）

ここですべての rd_data<?>信号をバンク2に割り当てます（図10）．

Design BrowserウィンドウでI/O Pinsフォルダを選択します．Design Object List - I/O Pinsウィンドウで，すべてのrd_dataグループを選択します．PACEがネットリストにあるデータ・バスを検出したときに，グループが自動的に作成されます．rd_dataのLoc列のセルの右端をクリックして表示されるドロップダウン・メニューから「BANK2」を選択します．

ドロップダウン・メニューには，以下のような選択肢もあります．

- T（Top）　　：チップ上端の二つのバンク
- L（Left）　　：チップ左端の二つのバンク
- TL（Top Left）：上端の左側のバンク

これらは，バンク番号がわからない場合に便利です．

2）フロアプラン

フロプランでは，指定のロケーションに特定のコンポーネントを割り当てます．ここでは，DCMコンポーネントをロケーションDCM_X0Y1に配置し，BRAM コンポーネントをブロック RAM ロケーションのX0Y3に配置してみます（図11）．

Design Browserウィンドウで「Global Logic」を選択します．次に，Design Object Listウィンドウでinst_my_dcm_DCM_INSTを選択し，Device ArchitectureウィンドウのDCM_X0Y1ロケーションにドラッグ＆ドロップします．

さらに，Design Object Listウィンドウで fifo_2048x8_inst_fifo_bramを選択し，BRAMロケーションX0Y3にドラッグ＆ドロップします．

「File」→「Save」を選択して，ピンの割り当てとDCM，BRAMの配置を保存します．そして，「File」→「Exit」を選択してPACEを終了します．

Project NavigatorのProcesses for Sourceウィンドウで，User Constraintsの下にある「Edit Constraints（Text）」をダブルクリックして，制約ファイルch_fifo.ucfの内容を確認します．

3）ピン配置（ピン番号指定）

制約ファイルを直接編集して，ピン配置を指定することができます（図12）．

Sources in Projectウィンドウでch_fifo.ucfが選択されていることを確認し，「Edit Constraints（Text）」をダブルクリックして，制約ファイルを開きます． almost_emptyを97番ピン，almost_fullを96番ピン，emptyを92番ピン，fullを91番ピンにする場合は，以下のテキストを追加し，セーブします．

```
NET "almost_empty"  LOC = "p97" ;
```

4

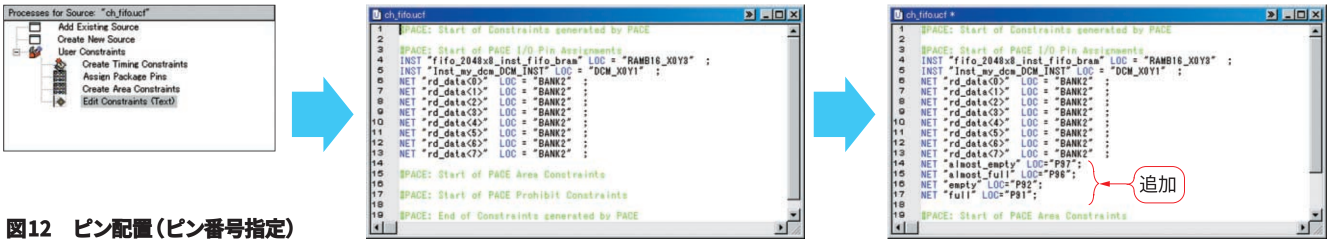


図12　ピン配置（ピン番号指定）

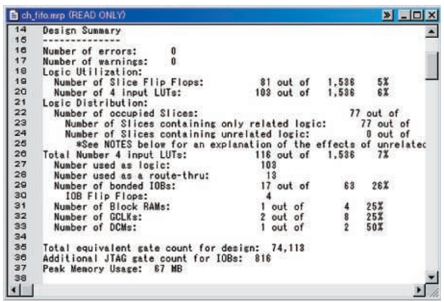


図13　使用リソースのレポート

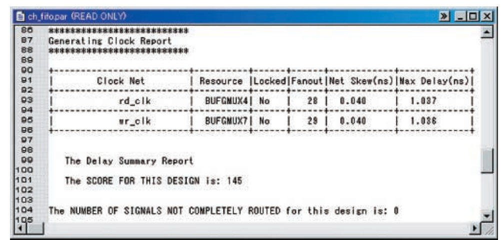


図14　タイミング・レポート

```
NET "almost_full"  LOC = "p96" ;
NET "empty"  LOC = "p92" ;
NET "full"  LOC = "p91" ;
```

4）配置配線の再実行

Sources in Projectソース・ウィンドウで，最上位ファイル（ch_fifo.vhd）が選択されていることを確認します．そして，Processes for Sourceウィンドウで，「Implement Design」をダブルクリックします．これで配置配線が再実行されます．論理合成は再実行しません．これは，Project Navigatorがネットリストには変更がないことを識別しているためです．

最後に，生成されたおもなレポートを確認します．「Implement Design」の［＋］をクリックして展開表示し，「Map」の下の「Map Report」をダブルクリックして，Design Summaryを確認します．ここには使用したリソースが報告されています（図13）．

「Place and Route」の「Place and Route Report」をダブルクリックして，Generating Clock Reportを確認します．ここにはクロックの最大スキューと最大遅延が報告されています（図14）．

● ビットストリームを生成し，コンフィグレーション

今までの手順で作成された設計データから，FPGAのプログラム用ファイルであるビットストリームを作成し，実際にFPGAをプログラミングします．

付属FPGA基板にLEDを拡張することで，FIFOのステータスを目で確認できます（full＝B15，empty＝A15，almost_full＝A16，almost_empty＝B16を使用する）．周辺回路がない場合は，コンフィグレーションの操作だけでも体験してください．

まず，Processes for Sourceウィンドウの「Generate Programming File」を選択します．次に，「Process」→「Properties...」を選択し，Process Propertiesウィンドウを開きます（図15）．Startup OptionsタブのFPGA Start-Up Clockを「JTAG Clock」にして，［OK］ボタンを押します．「Generate Programming File」をダブルクリックします．これでFPGAのプログラミング用のファイル（.bitファイル）が生成されます．

コンフィグレーション・ツール（Parallel Cable IVなど）をパソコンと付属FPGA基板に接続します．付属FPGA基板には電源もつなぎます．

「Generate Programming File」の［＋］をクリックして展開し，「Configure Device（iMPACT）」をダブルクリックします．これでダウンロード機能のiMPACTが立ち上がります．

Configure Devicesウィンドウでは，「Boundary-Scan Mode」を選択します．「Boundary-Scan Mode Selection」では「Automatically…」を選択します．これでJTAGでつ

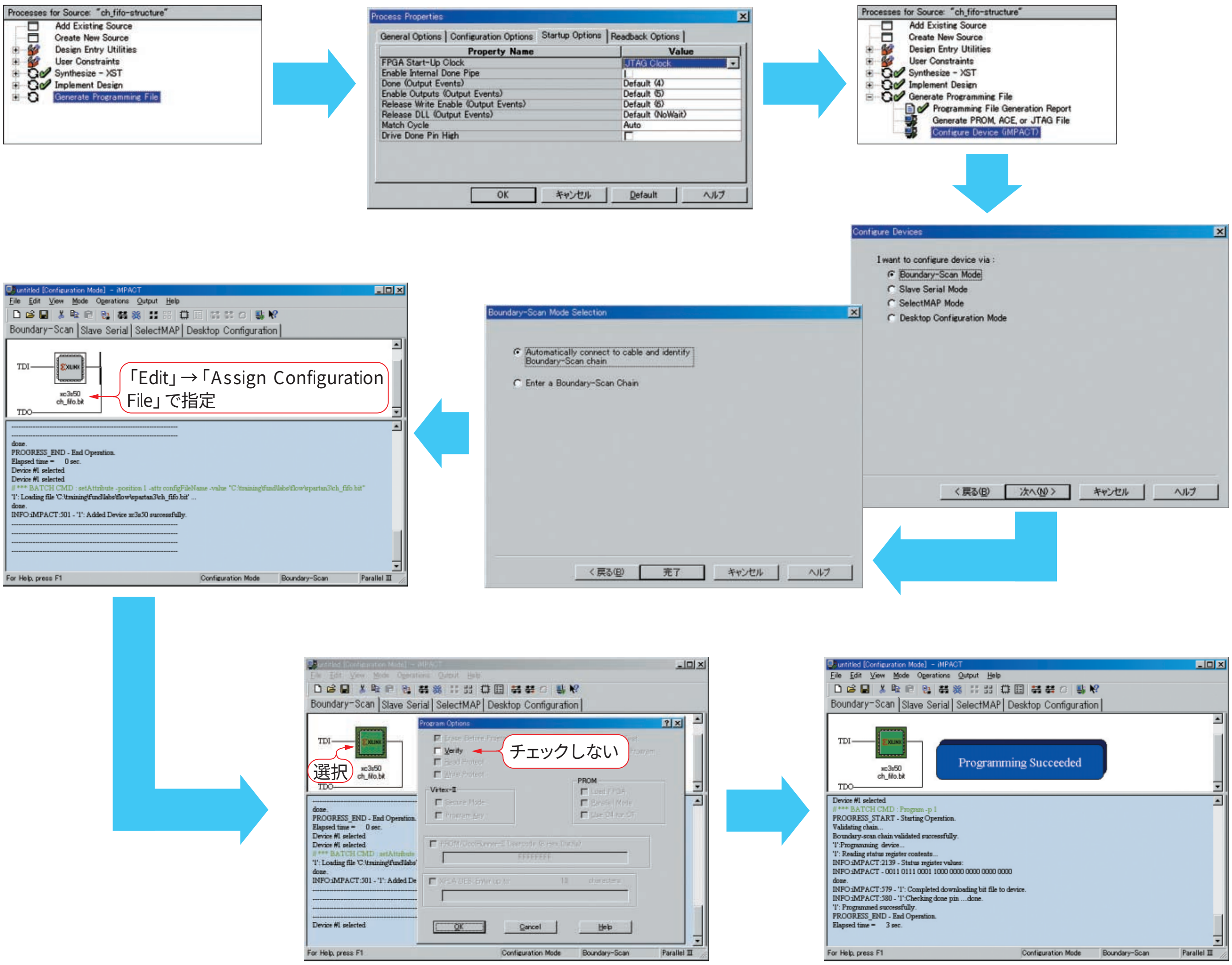


図15　ビットストリームの生成とコンフィグレーション

ながっているデバイスがグラフィックス表示されます．付属FPGA基板の場合は，XC3S50だけが表示されます（オプションのコンフィグレーション用ROMが搭載されている場合は，二つのデバイスが表示される）．

コンフィグレーションするデバイス（xc3s50）をクリックし，緑色になっていることを確認し，「Edit」→「Assign Con-figuration File」を選択します．そして，Assign New Configuration Fileウィンドウで「ch_fifo.bit」を選択します．画面のxc3s50の下がch_fifo.bitになっていることを確認します．

デバイスをクリックしたまま「Operations」→「Program」を選択すると，Program Optionsウィンドウが開きます．Verifyのチェックが入っていないことを確認し，［OK］ボタンをクリックすると，コンフィグレーションが始まります．「Programming Succeeded」と表示されたら，FPGAへのプログラミングは完了です．

なお，このch_fifo.bitは付属CD-ROMにも収録されています．これを使って，コンフィグレーションだけを実施することもできます．iMPACTを単体で立ち上げる場合は，Windowsのスタート・メニューより，「プログラム」→「Xilinx ISE 6」→「アクセサリ」→「iMPACT」を選択してください．

にしむら・けんじ
ザイリンクス(株)

編集部注：次号（本誌2005年2月号）の付属DVD-ROMには，Xilinx社のFPGA開発ツール「ISE Foundation 6.3i Evaluation Version」が収録される予定．